CLAIMS:

What is claimed is:

1. A method in a data processing system for serializing hardware reset requests in a software communication request queue in a processor card, said processor card processing software communication requests utilizing said queue in a serial order and processing hardware reset requests without utilizing said queue, said method comprising the steps of:

receiving a hardware reset request;

placing said hardware reset request in said queue; and

processing said hardware reset request from said queue in said serial order when all requests from said queue currently being serviced have completed being serviced.

2. The method according to claim 1, further comprising the steps of:

receiving said hardware reset request, said hardware reset request specifying one of a plurality of resource cards to reset;

said processor card being coupled to each one of said plurality of resource cards utilizing a single reset bus;

executing said hardware reset request by said processor card utilizing said reset line; and

resetting all of said plurality of resource cards simultaneously in response to said receipt said hardware

reset request specifying one of a plurality of resource cards to reset.

3. The method according to claim 2, further comprising the steps of:

receiving software communication requests and hardware reset requests;

placing said software communication requests and hardware reset requests in said queue in a serial order in which said software communication requests and hardware reset requests were received, said hardware reset requests being serialized within said queue with said software communication requests.

4. The method according to claim 2, further comprising the steps of:

looking at a next request in said serial order in said queue;

determining whether said next request is a hardware reset request;

in response to a determination that said next request is a hardware reset request, determining whether all of said plurality of resource cards have completed servicing of any pending software communication requests; and

waiting to execute said hardware reset request until all of said plurality of resource cards have completed servicing of any pending software communication requests.

5. The method according to claim 4, further comprising the steps of:

in response to a determination that all of said plurality of resource cards have completed servicing of any pending software communication requests, determining whether all of said plurality of resource cards have completed servicing of any pending sequence of multiple software communication requests; and

waiting to execute said hardware reset request until all of said plurality of resource cards have completed servicing of any pending sequence of multiple software communication requests.

6. The method according to claim 4, further comprising the steps of:

in response to a determination that all of said plurality of resource cards have completed servicing of any pending software communication requests, executing said hardware reset request utilizing said reset line; and

resetting all of said plurality of resource cards simultaneously.

7. The method according to claim 2, further comprising the steps of:

each one of said resource cards including a microcontroller and a memory; and

resetting said microcontroller in each one of said plurality of resource cards simultaneously in response to

said receipt said hardware reset request specifying one of a plurality of resource cards to reset.

8. The method according to claim 2, further comprising the steps of:

each one of said plurality of resource cards including a microcontroller, a memory, and synchronization bits;

utilizing said synchronization bits to maintain information about current servicing of software communication requests by each one of said plurality of resource cards;

resetting said microcontroller and said synchronization bits in each one of said plurality of resource cards simultaneously in response to said receipt said hardware reset request specifying one of a plurality of resource cards to reset; and

losing said information about current servicing of software communication requests by each one of said plurality of resource cards when said synchronization bits are reset.

9. A data processing system for serializing hardware reset requests in a software communication request queue in a processor card, said processor card processing software communication requests utilizing said queue in a serial order and processing hardware reset requests without utilizing said queue, said system comprising

a hardware reset request;

said queue for storing said hardware reset request; and

said queue for processing said hardware reset request from said queue in said serial order when all requests from said queue currently being serviced have completed being serviced.

10. The system according to claim 9, further comprising:

said hardware reset request specifying one of a plurality of resource cards to reset;

said processor card being coupled to each one of said plurality of resource cards utilizing a single reset bus;

said processor card for executing said hardware reset request utilizing said reset line; and

said processor card for resetting all of said plurality of resource cards simultaneously in response to said receipt said hardware reset request specifying one of a plurality of resource cards to reset.

11. The system according to claim 10, further comprising:

said queue for storing software communication requests and hardware reset requests in a serial order in which said software communication requests and hardware reset requests were received, said hardware reset requests being serialized within said queue with said software communication requests.

12. The system according to claim 10, further comprising:

said system including a CPU executing code for looking at a next request in said serial order in said queue;

said CPU executing code for determining whether said next request is a hardware reset request;

in response to a determination that said next request is a hardware reset request, said CPU executing code for determining whether all of said plurality of resource cards have completed servicing of any pending software communication requests; and

said processor card for waiting to execute said hardware reset request until all of said plurality of resource cards have completed servicing of any pending software communication requests.

13. The system according to claim 12, further comprising:

in response to a determination that all of said plurality of resource cards have completed servicing of any pending software communication requests, said CPU executing code for determining whether all of said plurality of resource cards have completed servicing of any pending sequence of multiple software communication requests; and

said processor card for waiting to execute said hardware reset request until all of said plurality of resource cards have completed servicing of any pending sequence of multiple software communication requests.

14. The system according to claim 12, further comprising:

in response to a determination that all of said plurality of resource cards have completed servicing of any pending software communication requests, said processor card for executing said hardware reset request utilizing said reset line; and

all of said plurality of resource cards being reset simultaneously.

15. The system according to claim 10, further comprising:

each one of said resource cards including a microcontroller and a memory; and

said microcontroller in each one of said plurality of resource cards being reset simultaneously in response to said receipt said hardware reset request specifying one of a plurality of resource cards to reset.

16. The system according to claim 10, further comprising:

each one of said plurality of resource cards including a microcontroller, a memory, and synchronization bits;

said synchronization bits for maintaining information about current servicing of software communication requests by each one of said plurality of resource cards;

said microcontroller and said synchronization bits being reset in each one of said plurality of resource

cards simultaneously in response to said receipt said hardware reset request specifying one of a plurality of resource cards to reset; and

said information about current servicing of software communication requests being lost by each one of said plurality of resource cards when said synchronization bits are reset.

17. A computer program product in a data processing system for serializing hardware reset requests in a software communication request queue in a processor card, said processor card processing software communication requests utilizing said queue in a serial order and processing hardware reset requests without utilizing said queue, said product comprising:

instruction means for receiving a hardware reset request;

instruction means for placing said hardware reset request in said queue; and

instruction means for processing said hardware reset request from said queue in said serial order when all requests from said queue currently being serviced have completed being serviced.

18. The product according to claim 17, further comprising:

instruction means for receiving said hardware reset request, said hardware reset request specifying one of a plurality of resource cards to reset;

said processor card being coupled to each one of said plurality of resource cards utilizing a single reset bus;

instruction means for executing said hardware reset request by said processor card utilizing said reset line; and

instruction means for resetting all of said plurality of resource cards simultaneously in response to said receipt said hardware reset request specifying one of a plurality of resource cards to reset.

19. The product according to claim 18, further comprising:

instruction means for receiving software communication requests and hardware reset requests;

instruction means for placing said software communication requests and hardware reset requests in said queue in a serial order in which said software communication requests and hardware reset requests were received, said hardware reset requests being serialized within said queue with said software communication requests.

20. The product according to claim 18, further comprising:

instruction means for looking at a next request in said serial order in said queue;

instruction means for determining whether said next request is a hardware reset request;

in response to a determination that said next request is a hardware reset request, instruction means for determining whether all of said plurality of resource cards have completed servicing of any pending software communication requests; and

instruction means for waiting to execute said hardware reset request until all of said plurality of resource cards have completed servicing of any pending software communication requests.

21. The product according to claim 20, further comprising:

in response to a determination that all of said plurality of resource cards have completed servicing of any pending software communication requests, instruction means for determining whether all of said plurality of resource cards have completed servicing of any pending sequence of multiple software communication requests; and

instruction means for waiting to execute said hardware reset request until all of said plurality of resource cards have completed servicing of any pending sequence of multiple software communication requests.

22. The product according to claim 20, further comprising:

in response to a determination that all of said plurality of resource cards have completed servicing of any pending software communication requests, instruction means for executing said hardware reset request utilizing said reset line; and

instruction means for resetting all of said plurality of resource cards simultaneously.

23. The product according to claim 18, further comprising:

each one of said resource cards including a microcontroller and a memory; and

instruction means for resetting said microcontroller in each one of said plurality of resource cards simultaneously in response to said receipt said hardware reset request specifying one of a plurality of resource cards to reset.

24. The product according to claim 18, further comprising:

each one of said plurality of resource cards including a microcontroller, a memory, and synchronization bits;

instruction means for utilizing said synchronization bits to maintain information about current servicing of software communication requests by each one of said plurality of resource cards;

instruction means for resetting said microcontroller and said synchronization bits in each one of said plurality of resource cards simultaneously in response to said receipt said hardware reset request specifying one of a plurality of resource cards to reset; and

instruction means for losing said information about current servicing of software communication requests by

each one of said plurality of resource cards when said synchronization bits are reset.